

Docket No. AUS920030556US1

CLAIMS:

What is claimed is:

1. A method, in a data processing system, for optimizing runtime execution of a computer program, comprising:

modifying performance profile data accumulated during a trace of a computer program execution to include annotations based on the occurrence of one or more events, wherein the one or more events occur based on hardware counter values and performance indicators associated with one or more portions of the computer program;

providing the annotated performance profile data to a computer program compiler; and

identifying one or more optimizations to the runtime execution of the computer program that may be performed based on the performance profile data and the annotations.

2. The method of claim 1, further comprising:

outputting an indication of the identified one or more optimizations to a user.

3. The method of claim 2, further comprising:

receiving a user input selecting at least one of the one or more optimizations to be implemented; and

Docket No. AUS920030556US1

implementing the selected at least one of the one or more optimizations to generate an optimized computer program.

4. The method of claim 3, wherein the one or more optimizations include at least one of an optimization to the instruction paths of the computer program, optimization of time spent during an initial application load of the computer program, and an optimization to storage of instructions or data in a cache or memory.

5. The method of claim 1, further comprising:
automatically implementing at least one of the one or more optimizations to the runtime execution of the computer program to generate an optimized computer program.

6. The method of claim 3, wherein the one or more optimizations includes optimizing instructions paths of the computer program at branch points such that a more contiguous execution of instructions within the computer program is achieved.

7. The method of claim 3, wherein the one or more optimizations includes optimizing storage of instructions or data in a cache so that portions of a cache line that is falsely shared are stored in the cache on different cache lines.

Docket No. AUS920030556US1

8. The method of claim 2, wherein the indication is a graphical user interface through which a user may select respective ones of the one or more optimizations that are to be implemented with code of the computer program during compilation of the computer program.

9. The method of claim 3, wherein the selected at least one of the one or more optimizations is implemented by a compiler during compilation of the computer program.

10. The method of claim 5, wherein the at least one of the one or more optimizations is automatically implemented by a compiler during compilation of the computer program.

11. A computer program product in a computer readable medium for optimizing runtime execution of a computer program, comprising:

first instructions for modifying performance profile data accumulated during a trace of a computer program execution to include annotations based on the occurrence of one or more events, wherein the one or more events occur based on hardware counter values and performance indicators associated with one or more portions of the computer program;

second instructions for providing the annotated performance profile data to a computer program compiler;
and

third instructions for identifying one or more optimizations to the runtime execution of the computer

Docket No. AUS920030556US1

program that may be performed based on the performance profile data and the annotations.

12. The computer program product of claim 11, further comprising:

fourth instructions for outputting an indication of the identified one or more optimizations to a user.

13. The computer program product of claim 12, further comprising:

fifth instructions for receiving a user input selecting at least one of the one or more optimizations to be implemented; and

sixth instructions for implementing the selected at least one of the one or more optimizations to generate an optimized computer program.

14. The computer program product of claim 13, wherein the one or more optimizations include at least one of an optimization to the instruction paths of the computer program, optimization of time spent during an initial application load of the computer program, and an optimization to storage of instructions or data in a cache or memory.

15. The computer program product of claim 11, further comprising:

fourth instructions for automatically implementing at least one of the one or more optimizations to the

Docket No. AUS920030556US1

runtime execution of the computer program to generate an optimized computer program.

16. The computer program product of claim 13, wherein the one or more optimizations includes optimizing instructions paths of the computer program at branch points such that a more contiguous execution of instructions within the computer program is achieved.

17. The computer program product of claim 13, wherein the one or more optimizations includes optimizing storage of instructions or data in a cache so that portions of a cache line that is falsely shared are stored in the cache on different cache lines.

18. The computer program product of claim 12, wherein the indication is a graphical user interface through which a user may select respective ones of the one or more optimizations that are to be implemented with code of the computer program during compilation of the computer program.

19. The computer program product of claim 13, wherein the selected at least one of the one or more optimizations is implemented by a compiler during compilation of the computer program.

Docket No. AUS920030556US1

20. An apparatus for optimizing runtime execution of a computer program, comprising:

means for modifying performance profile data accumulated during a trace of a computer program execution to include annotations based on the occurrence of one or more events, wherein the one or more events occur based on hardware counter values and performance indicators associated with one or more portions of the computer program;

means for providing the annotated performance profile data to a computer program compiler; and

means for identifying one or more optimizations to the runtime execution of the computer program that may be performed based on the performance profile data and the annotations.